Software Signature Quality

Whitelist Coverage and Source-Authenticity

707 SW Washington, 7th Floor
Portland, Oregon 97205
Telephone: 503.227.2207
www.signacert.com

Executive Summary

Today, there is an increasing need for enterprise IT organizations to know and actively measure the integrity-state of all devices that are within the corporate domain. When application, configuration, and system compliance are actively verified, an enterprise benefits by reducing TCO (total cost of operations)through increased stability and reduced MTTR (mean time to repair), increasing overall availability of their network endpoints. Visibility into the aggregate compliance of all devices supporting a given business service is key to ensuring maximum availability of that service.

One highly effective methodology for determining device compliance to a known and expected configuration is to validate its contents against a repository of identified authentic individual data elements (such as file signatures), called a whitelist. The use of cryptographic file hashing techniques optimizes this method of comparison and provides the most efficient way to positively identify which data elements on a particular device are known. When employed as an IT best practice, this method can dramatically increase visibility into the overall state of each device and quickly identify critical problems that may affect overall device and network stability and availability.

In order for these methods to be most effective, the whitelists used for comparison ***must*** have the following critical attributes:

- Contain existing software signatures and metadata for the broadest range of commercially available software that is relevant to an enterprise's specific needs
- Contain only source authentic signatures and metadata from verifiable sources, ensuring that only the highest quality information is ever used for comparison
- Continuously synchronizes the software signatures and metadata it contains against the release schedules of major vendors and software publishers
- Provide simple and flexible mechanisms for extending the repository with software signatures for internally developed and customized applications which are found within each unique enterprise environment
- Provides flexible methods of organizing software signatures that make sense for each unique IT organization, i.e. by platform, server type, gold disk image, etc.

Using a known provenance whitelist, IT organizations will:

- Prove regulatory and internal policy compliance
- Increase IT operational efficiency
- Improve application lifecycle stability

Introduction

Enterprise IT organizations have a growing need to know at all times the configuration and codebase of the systems for which they are responsible. IT administrators who know exactly what software is installed and capable of executing within their enterprise possess a level of operational visibility that is core to the purpose and function of their organization.

One highly effective technique for gaining maximum visibility and understanding is through the use of whitelist methods. By giving IT departments the capability to determine which applications, configurations, files, and settings on a system are known and authentic, the whitelist methods improve system reliability and predictability by identifying compliance with a desired state. Having visibility into the specific deviations that caused the device(s) to a non-compliant state, provides IT professionals with the information they need to be proactive in preventing downtime. When devices fail, the time to diagnose and remediate can be significantly reduced by verifying the devices components and configurations against the desired reference to pin-point which elements are out of compliance.

Creating and maintaining a whitelist is a challenging and resource-intensive process. To be trusted and effective, whitelist signatures and other detailed information for each software product used within an enterprise must be collected, vetted, and stored. Since every enterprise has custom and proprietary software specific to their unique business, a successful and complete whitelist solution provides the tools and mechanisms necessary to automate and extend whitelist coverage to include signatures and metadata information for those custom and proprietary applications. Once the whitelist is created, it must be continuously synchronized with updated information from major software vendors and each update to enterprise-developed software and configurations.

By far the two most critical aspects of an effective whitelist solution are signature quality and coverage. Each is essential in providing the means to determine if the software signatures on a device represent the software as it was authored by the manufacturer. Without both, unknown, untested, and unauthorized code can be present in the enterprise, thus increasing risk, while decreasing the device's predictability.

By starting with a foundation of the highest quality source authentic signatures and augmenting those with custom signatures, an enterprise is assured that it has the most extensive, accurate, and trusted whitelist available.

Signature Quality in Whitelists and why it Matters

When creating a whitelist for use in measuring enterprise integrity, all software signatures that comprise the whitelist must have known provenance. Signatures that do not have such attributes are of limited or no value for use in any measurement or compliance verification. Any whitelist created without a strong basis of authenticity greatly diminishes

value gained by its use. When source authenticity is absent, it can be shown that there exists an even greater probability that bad, or even malicious software, can be authorized for use within the enterprise. In this way, the very data you are being asked to trust is drawn into question.

Therefore, the quality (or source-authenticity) of each signature in the whitelist must be clearly indicated and based upon a rating system that indicates its handling (chain of custody) and that is directly related to the location in the software supply-chain at which it is captured. Additionally, other metrics can be applied to improve handling and provenance and thereby improve this rating. For commercially available software applications, the point of signature capture should be integrated with, or as close as possible to, the software release process. The further a software product travels from its original point of release, the lower the confidence level will be in the authentic nature of the product itself. The confidence level is independent of any tamper-resistant delivery mechanism that may be deployed (CD/DVD, digital signatures, code signing, etc).

Relying upon a whitelist supplier who mitigates all of these concerns and risks, that aggregates signatures across a broad range of commercial and open source software providers, and provides an enterprise-ready whitelist capability, significantly simplifies the process for creating the foundation for extending highly valuable whitelist methods across any IT infrastructure.

Sample Use Case: Standard Build Compliance

When a device is initially provisioned for use within the enterprise, its contents are typically set to include only approved applications, both commercial and custom, according to corporate policy. This can be accomplished by imaging the machine from a known gold master image, or in some cases this is provided as a service by the PC platform manufacturer. At that moment in time, the exact contents of a device are known and it is considered compliant. As new software and updates to existing products are installed, the device begins to drift from the original state. Eventually, without proper measurement and visibility, the device may drift so far that it becomes unstable resulting in calls to the help desk or even unplanned system downtime. The source of these changes can be from ostensibly good sources such as vendor-issued patches and IT updates, but also from undesirable sources such as human tampering or changes caused by malware.

Today, when a device becomes unstable, the typical approach taken by many IT administrators is to erase the contents of the device and re-image the platform back to the original gold master. This reactive approach consumes considerable time and IT resources, not to mention the disruption and loss of productivity for the user whose device is thus unavailable. Also, it assumes that the changes made to the platform are of no value to the end user's productivity. Finally, it removes any ability for the IT professional to do a root cause analysis in order to prevent similar future changes from causing unplanned downtime.

A more proactive approach is needed to identify changes as a device's contents change over time. Conducting periodic checks against the device to detect drift is an example of this new approach which implements regular re-validation after initial provisioning. This new method provides an initial baseline rooted in the quality of the whitelist which attests to the authenticity of the device's contents, together with the ability to take preventative measures in a timely fashion by applying appropriate maintenance before systems become unstable or otherwise compromised. A key element in this new approach is to ensure the whitelist itself is continually updated with source-authentic signatures of the most recent versions of the relevant software.

Summary

As has been illustrated in this paper, there are a number of factors that contribute to the notion of quality of signatures and its affect on the viability of using whitelist methods in the enterprise.

Creating an effective whitelist solution requires an initial base of relevant signatures which cover the broadest range of software that are important to a specific enterprise. Having the ability to extend the signature base with timely updates from specified vendors via a reliable source who can attest to the authenticity of those signatures, as well as add custom and proprietary signatures and information makes the whitelist an accurate and invaluable asset. The accuracy of the whitelist information is paramount to any enterprise's ability to make assertions about the integrity of their network and systems.

The benefits of a whitelist approach are multi-fold: system drift from a desired state can be detected and addressed before problems occur, thereby improving system stability and reliability. When systems do fail, whitelist-based solutions speed problem diagnosis and remediation by validating which system components are in a desired state and pin-pointing missing or altered files.

The final result of implementing a whitelist rooted in the foundation of quality signatures is significant improvement in existing security and management for enterprise IT. Changes from authentic state can be detected and proactively addressed prior to system failure, thereby increasing system reliability, reducing costs, improving security, compliance and uptime.

About SignaCert

SignaCert is the leading provider of next-generation IT compliance solutions allowing organizations to rapidly achieve and prove continuous compliance for the systems that deliver critical business services. SignaCert's patented technology can be quickly deployed and provides immediate visibility into the actual state of IT infrastructure. The SignaCert architecture is designed to seamlessly integrate with existing change processes and continuously monitor critical business services without disruption.

Founded in 2004 by 34-year IT security and compliance industry veteran Wyatt Starnes, SignaCert has assembled a world class team of industry leaders with hands-on IT experience for its executive team, board of directors, and advisory board. SignaCert's customers span a wide variety of industries, including financial services, government, and healthcare.

For more information visit: www.signacert.com.

Glossary

Term	Definition
Coverage	The percentage of measured files that can be identified on a given device and/or the enterprise.
Provenance	Provenance means that individual data elements (such as the files that make up an application) are traceable and proven to originate from the source publisher or manufacturer.
Reference	The unique composition of cryptographic hash, and/or attributes of a collection of files, settings, and configurations. Typically a software application (i.e. Firefox) would be a single product reference.
Source-Authenticity Score or SAS	The numeric value that represents the confidence level in the source-authenticity of a given software signature.
Whitelist	A list of approved and/or known items in a reference set